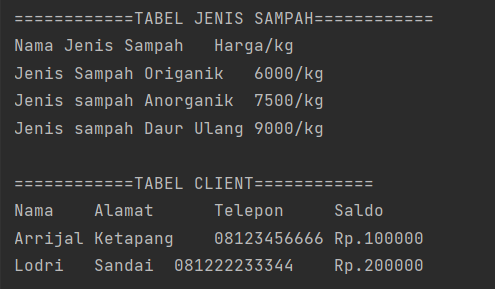
**SOAL PRAKTIK UAS**

**PEMROGRAMAN BERORIENTASI OBJEK**

**APLIKASI BANK SAMPAH**



**DISUSUN OLEH :**

**Arrijal Firmansyah**

**2100018350**

**Kelas G**

**Link Github:** [**https://github.com/ArrijalFirmansyah/TA\_2100018350\_Aplikasi-Bank-Sampah/tree/main/src**](https://github.com/ArrijalFirmansyah/TA_2100018350_Aplikasi-Bank-Sampah/tree/main/src)

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS TEKNOLOGI INDUSTRI**

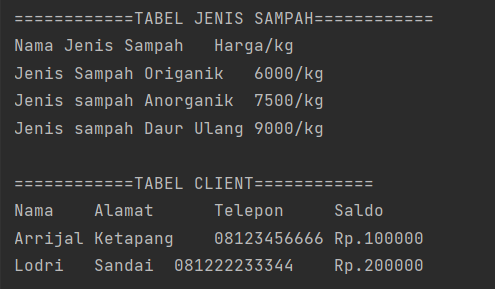
**UNIVERSITAS AHMAD DAHLAN**

**DESEMBER 2022**

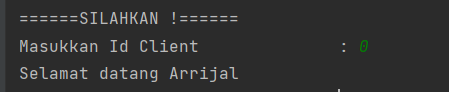
1. Ruang lingkup/deskripsi dari projek

Program aplikasi ini dirancang untuk nantinya akan digunakan untuk menukarkan barang-barang yang tidak terpakai (sampah) Organik, daur ulang dan sampah Anorganik. Berdasarkan kriteria sampah yang ditukarkan maka user akan mendapatkan uang/hadiah yang berbeda-beda juga tergantung dari berat (kg) dan kriteria jenis sampah yang ditukarkan.

1. Daftar seluruh spesifikasi aplikasi sesuai studi kasus.
2. Pada pertamakali menjalankan user akan mendapatkan TABEL JENIS SAMPAH dan TABEL CLIENT.



1. Setelah mendapatkan kedua tabel user akan diminta untuk menginputkan Id client.



1. Setelah menginput Id client user akan diminta untuk memasukkan kode jenis sampah. Disini ada 3 pilihan untuk 0 = jenis sampah organic, 1=Jenis sampah anorganik dan 2=Jenis sampah daur ulang.

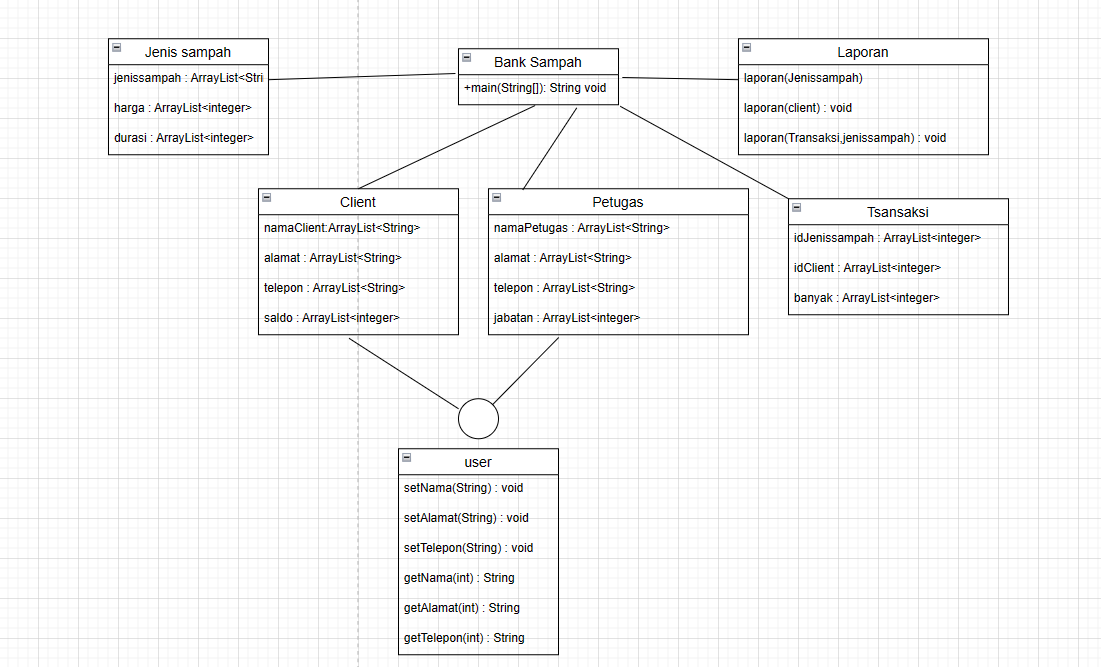


1. Setelah itu user akan di minta memasukkan banyak sampah yang di bawa :

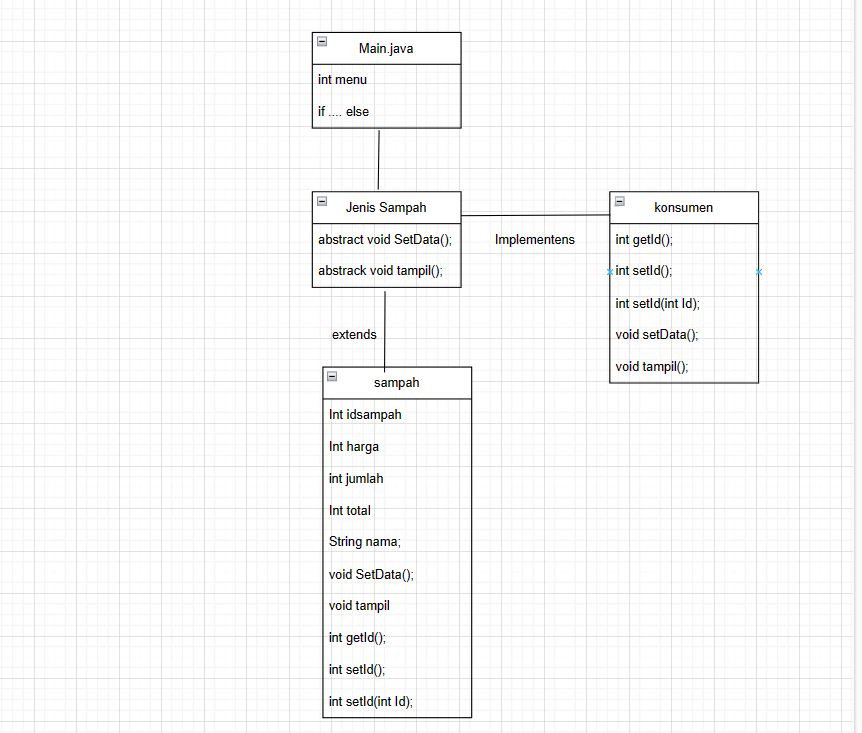


1. Dan proses penginputan selesai dengan memasukkan nomor 99
2. Setelah itu lanjut ke proses transaksi penukaran yang nani akan berisi jenis sampah, banyak(kg), harga(kg) dan jumlah semuahnya.
3. Kemudian selesai.
4. Diagram proses aplikasi (bisa DFD/UML/flowchart/ proses bisnis, atau diagram lainnya yang digunakan untuk menjelaskan jalannya program.

* Tampilan di Profosal.



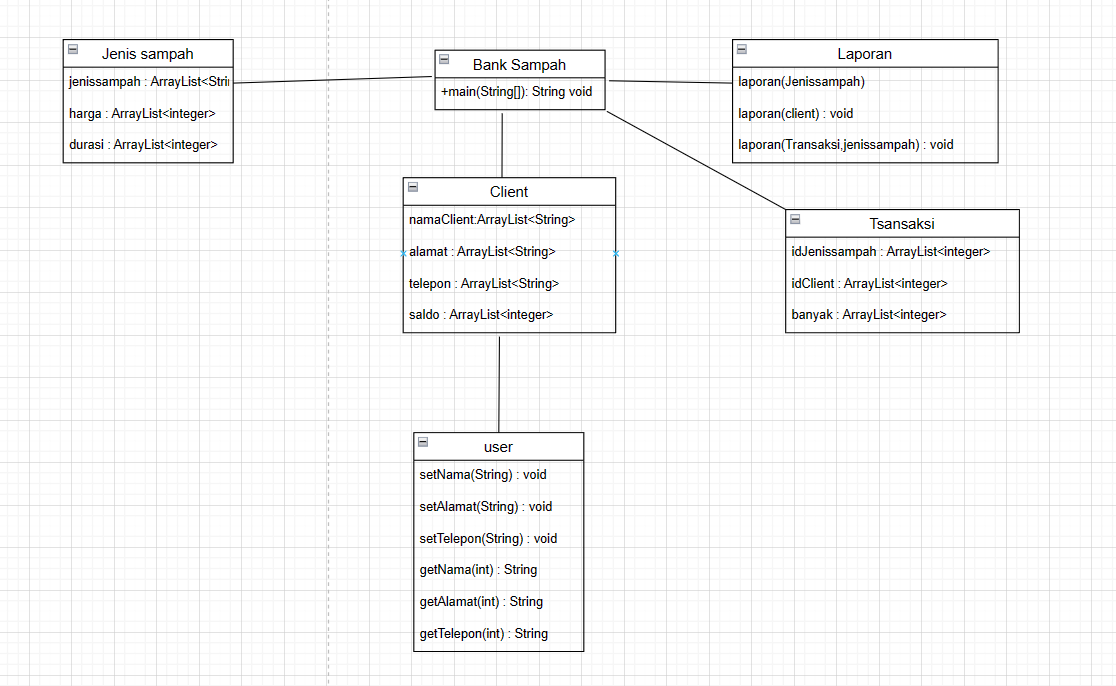
* Diagram mengalami perubahan.



* Tampilan diagram Final.

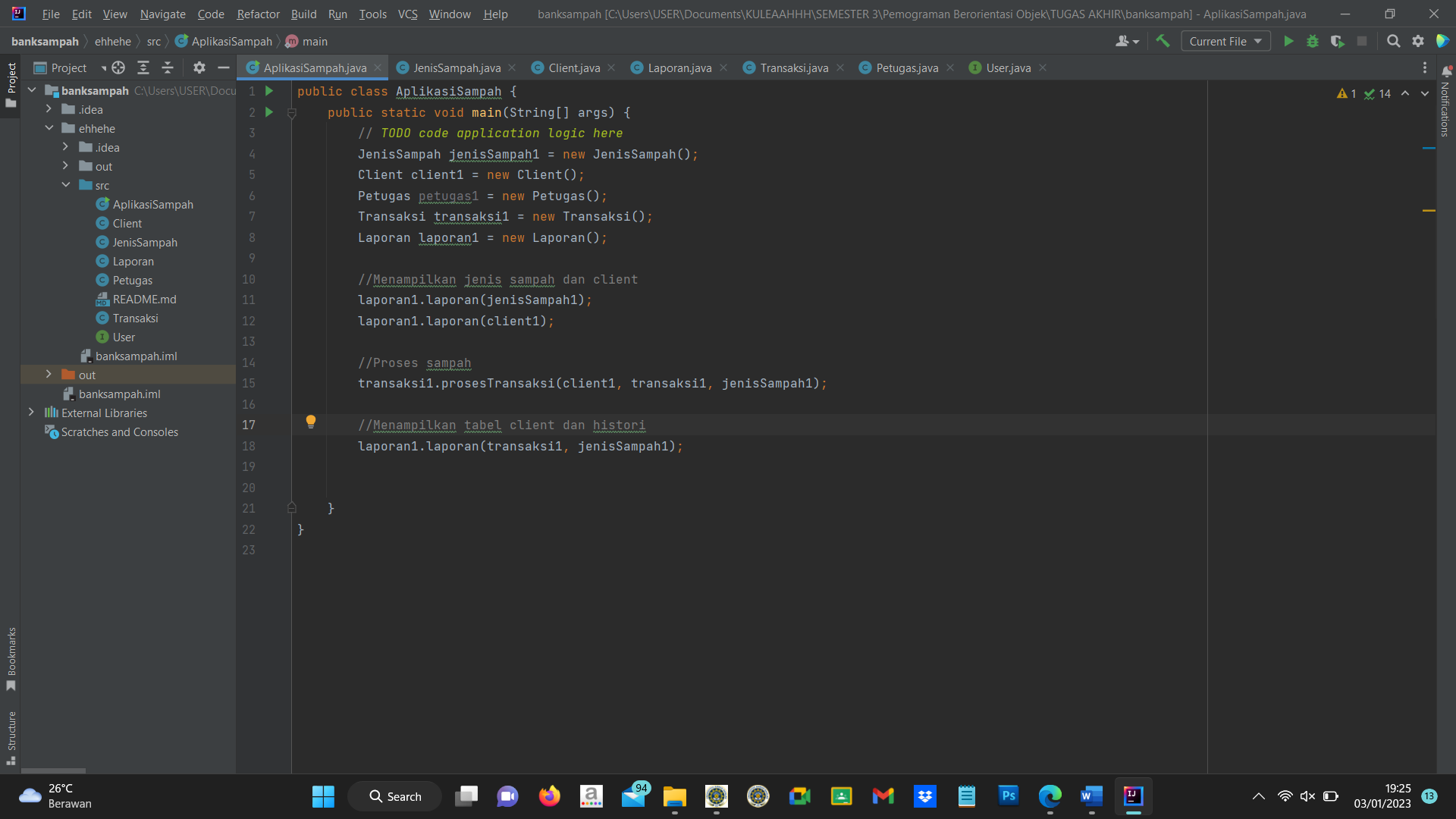
Pada Laporan akhir(Final) ini saya Kembali ke konsep awal yang sebelumnya saya menggunakan konsep baru namun tidak terealisasikan.

Berikut diagram FINAL :

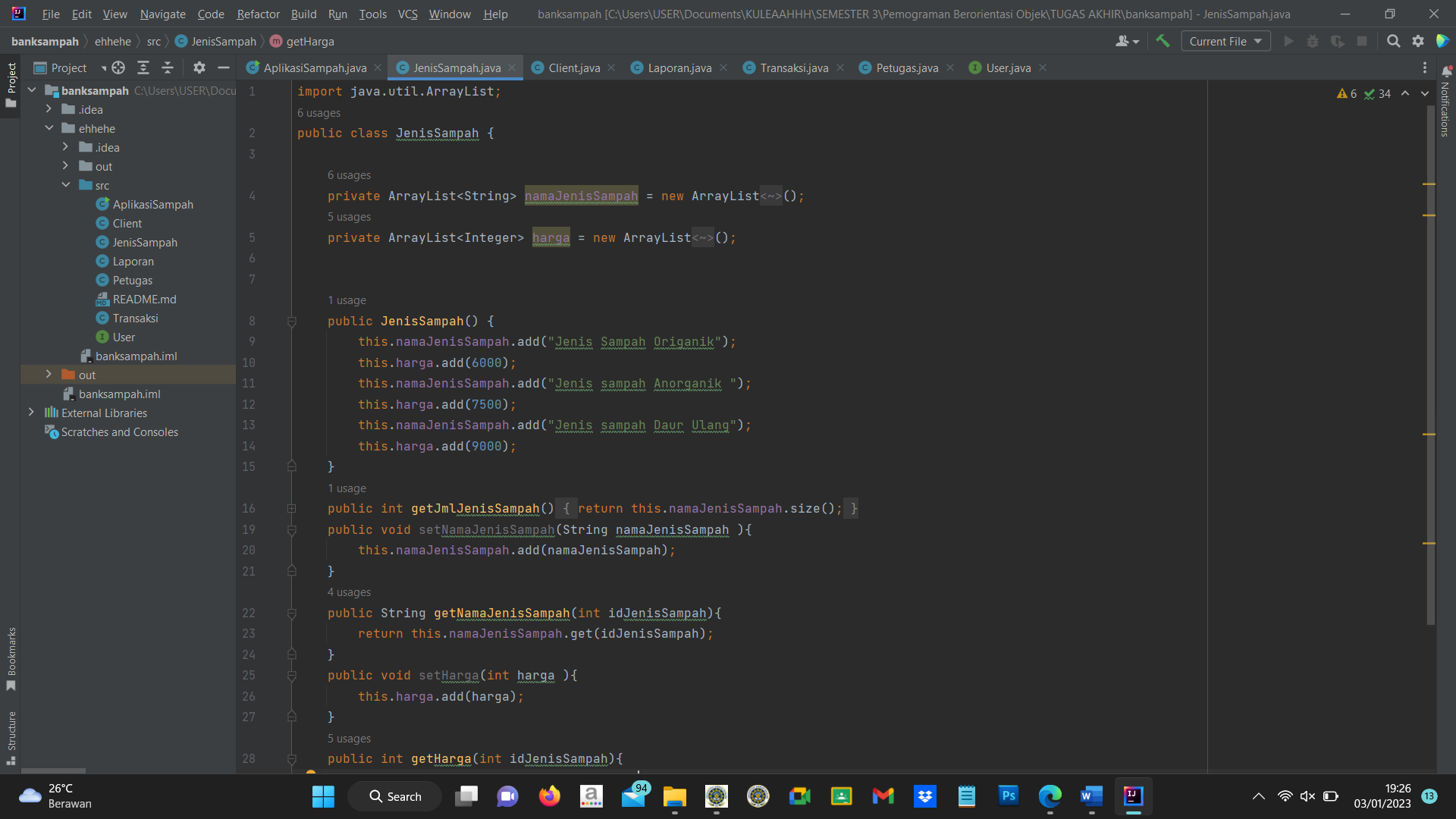


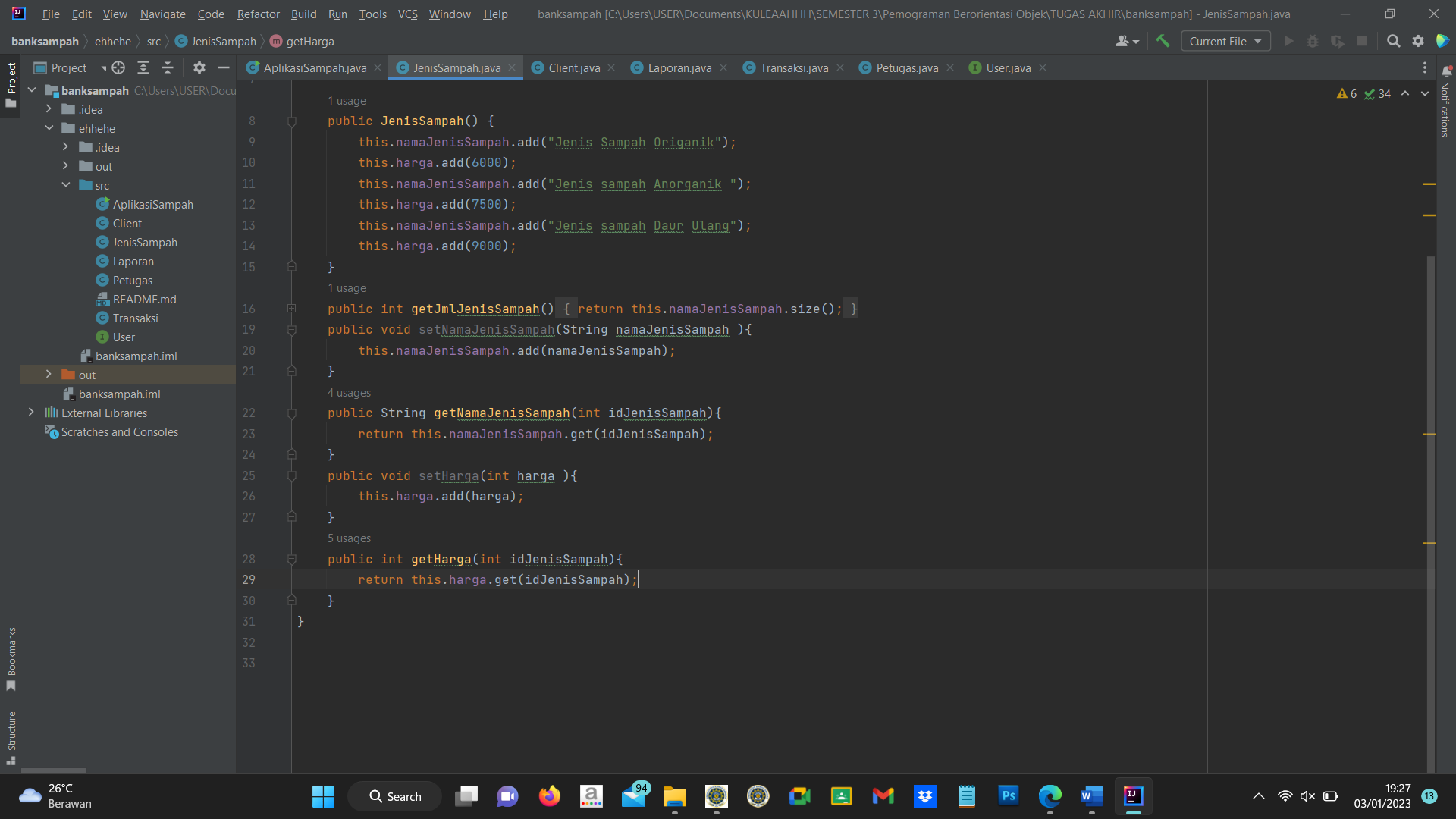
1. Penjelasan koding/skrip program.

* AplikasiSampah.java (main)

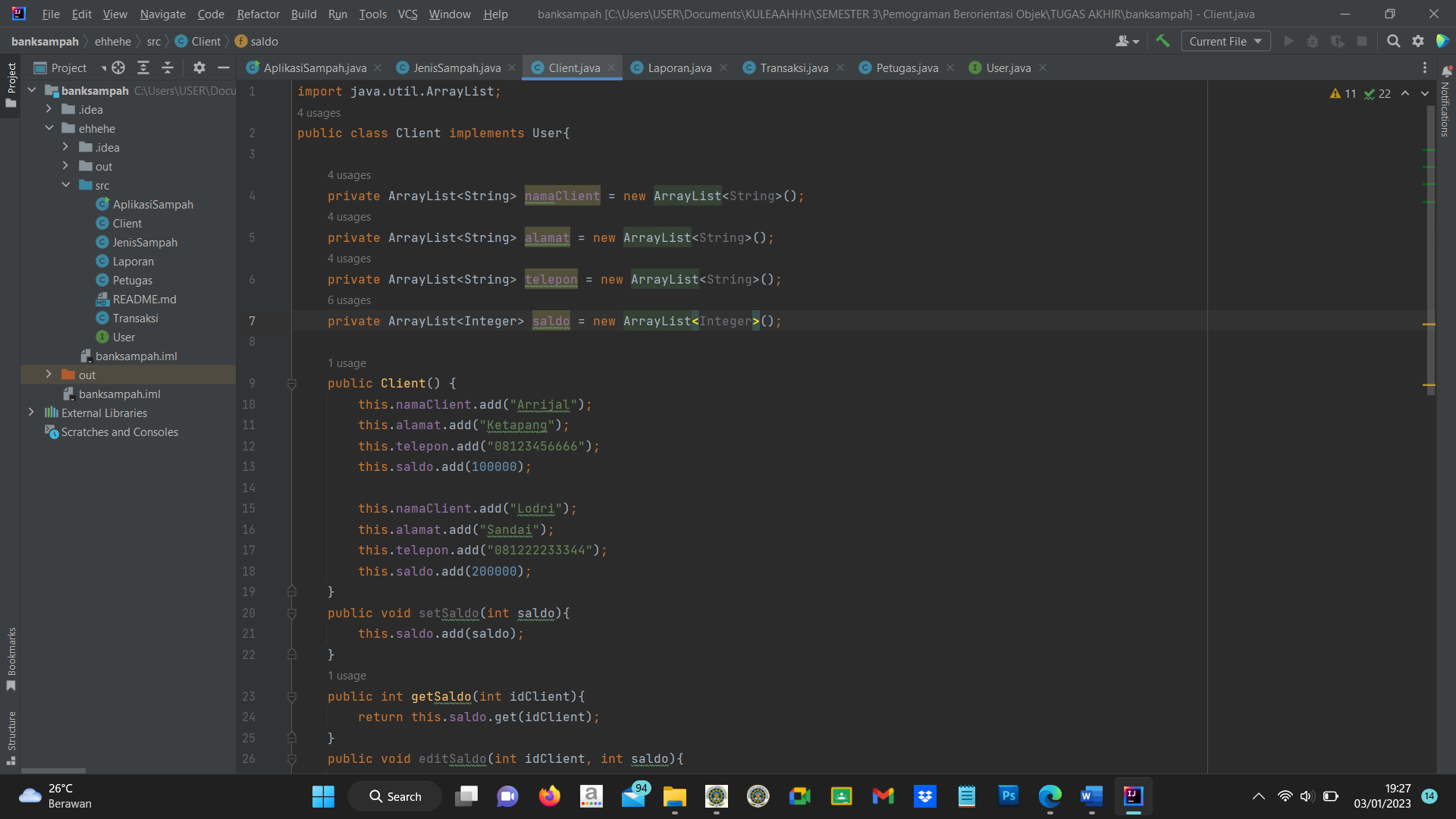


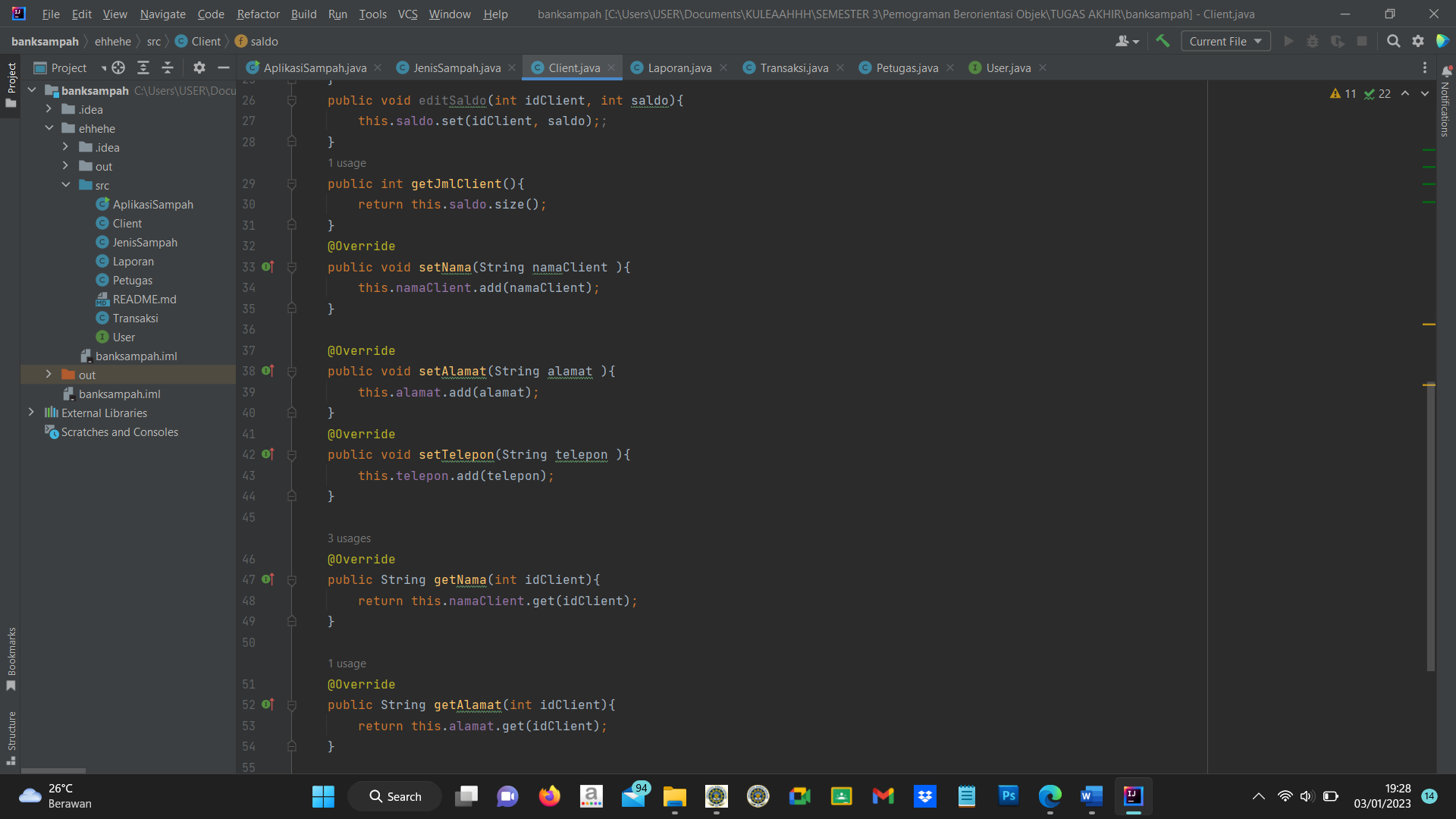
* JenisSampah.java

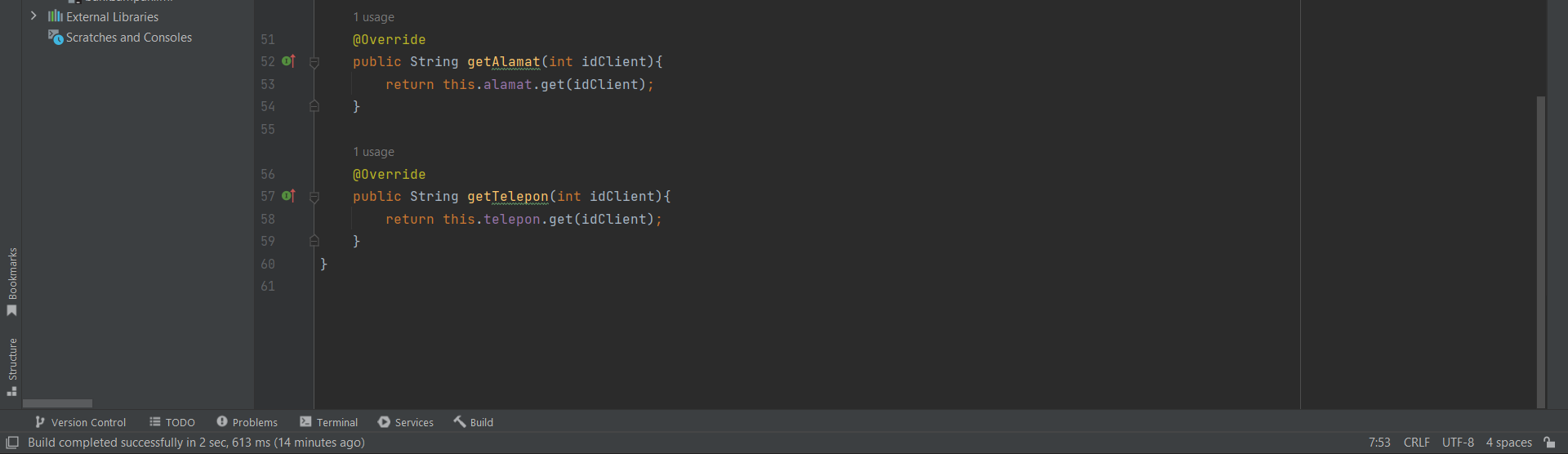




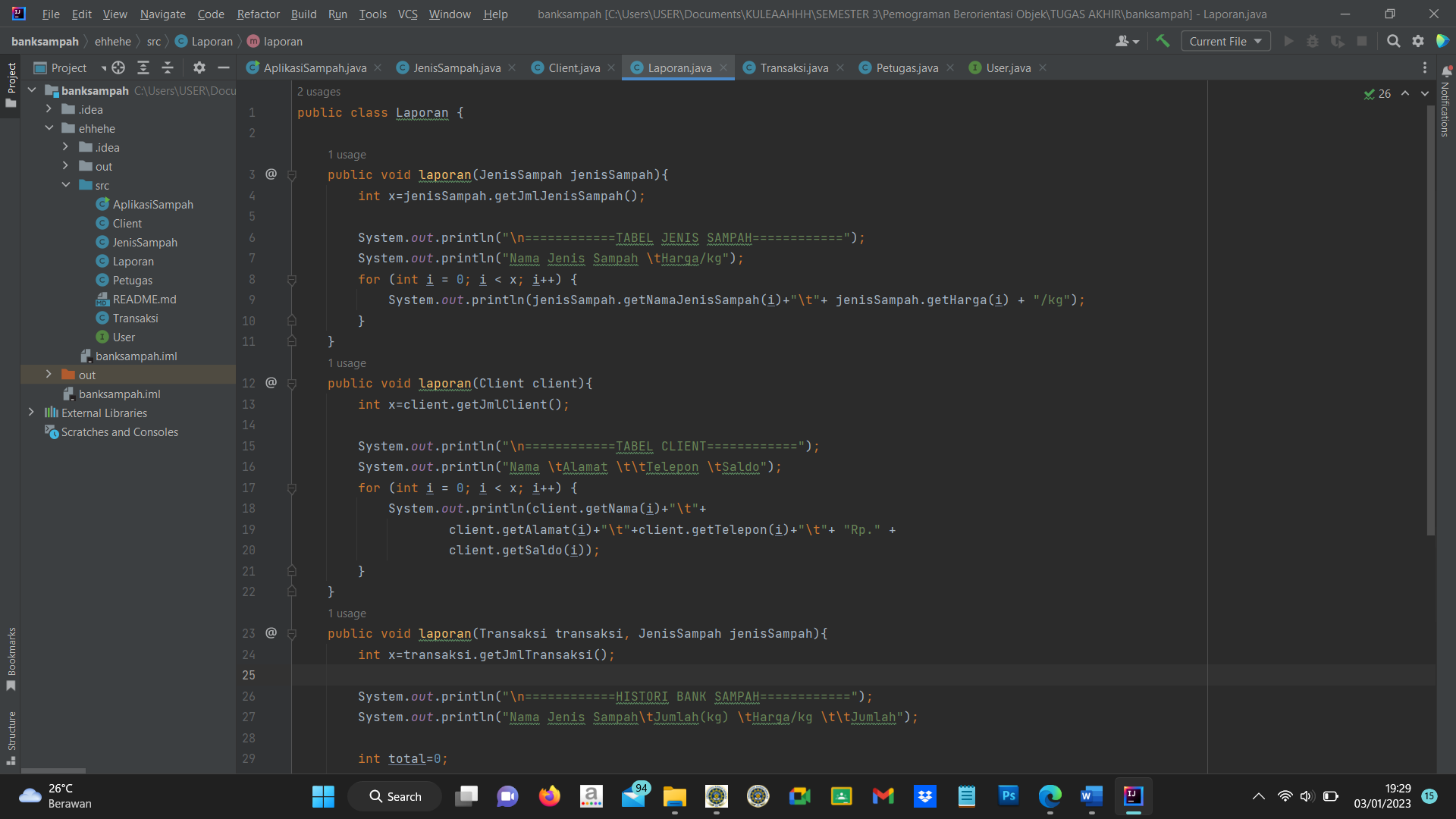
* Client.java

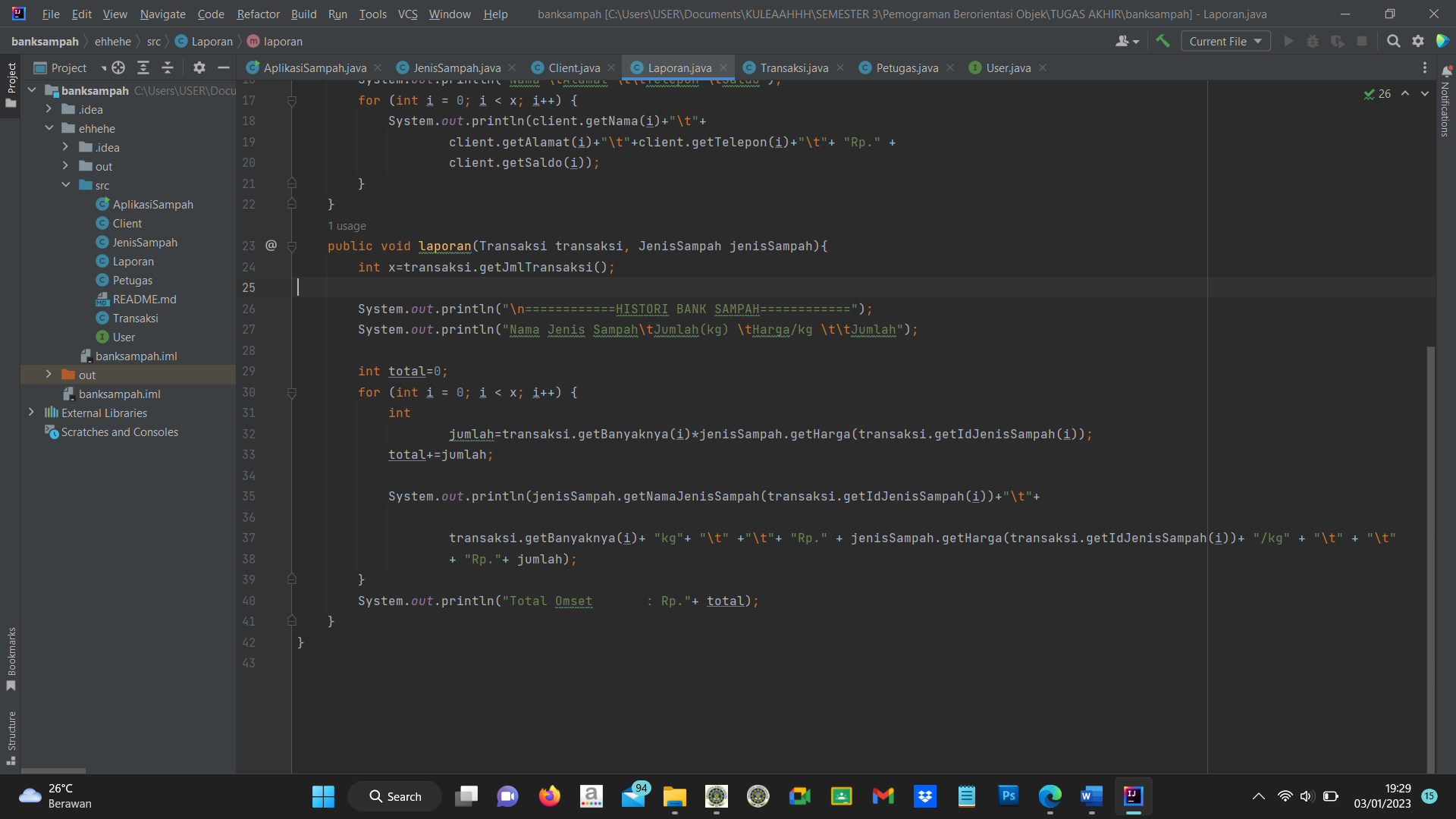




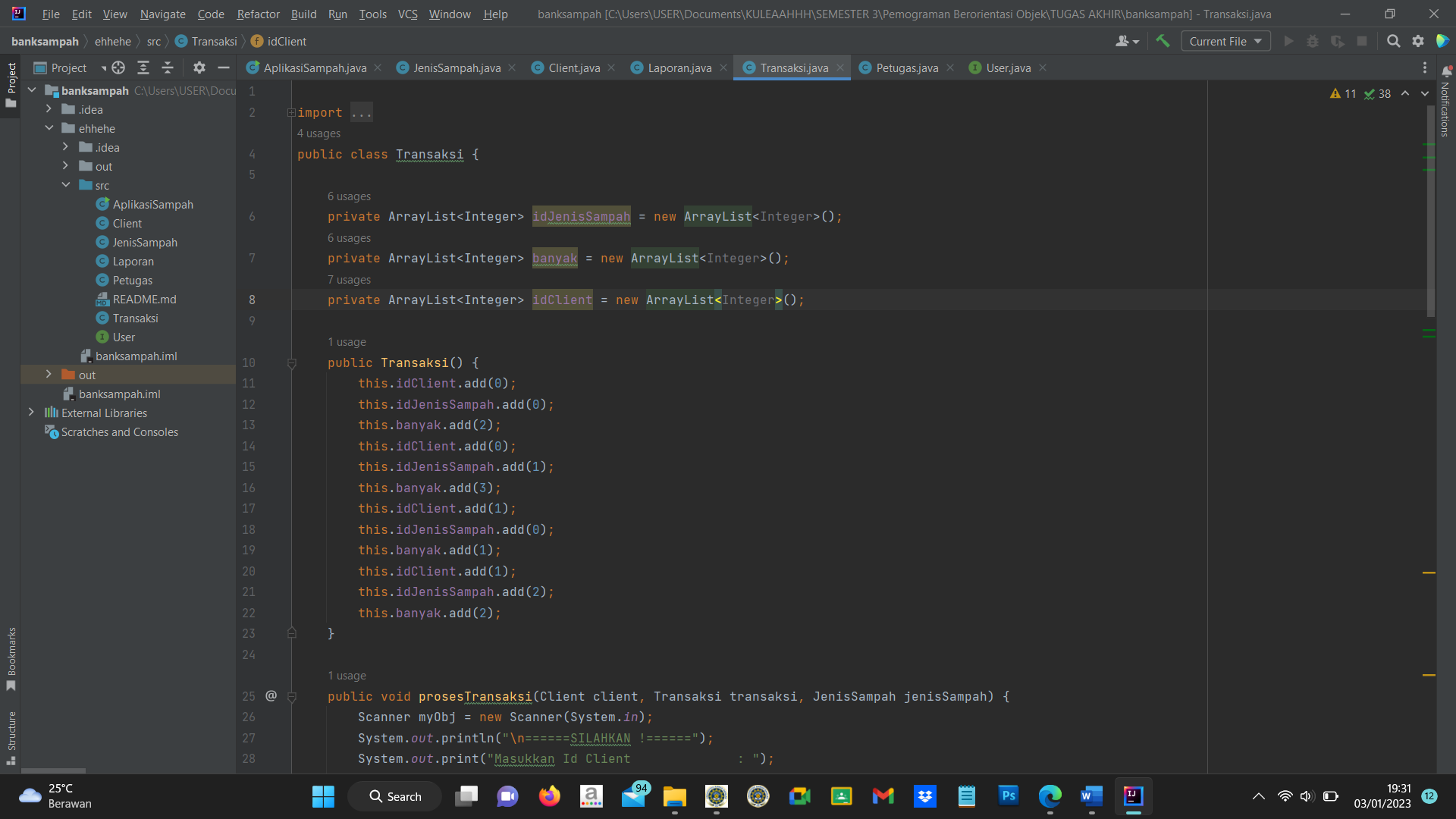


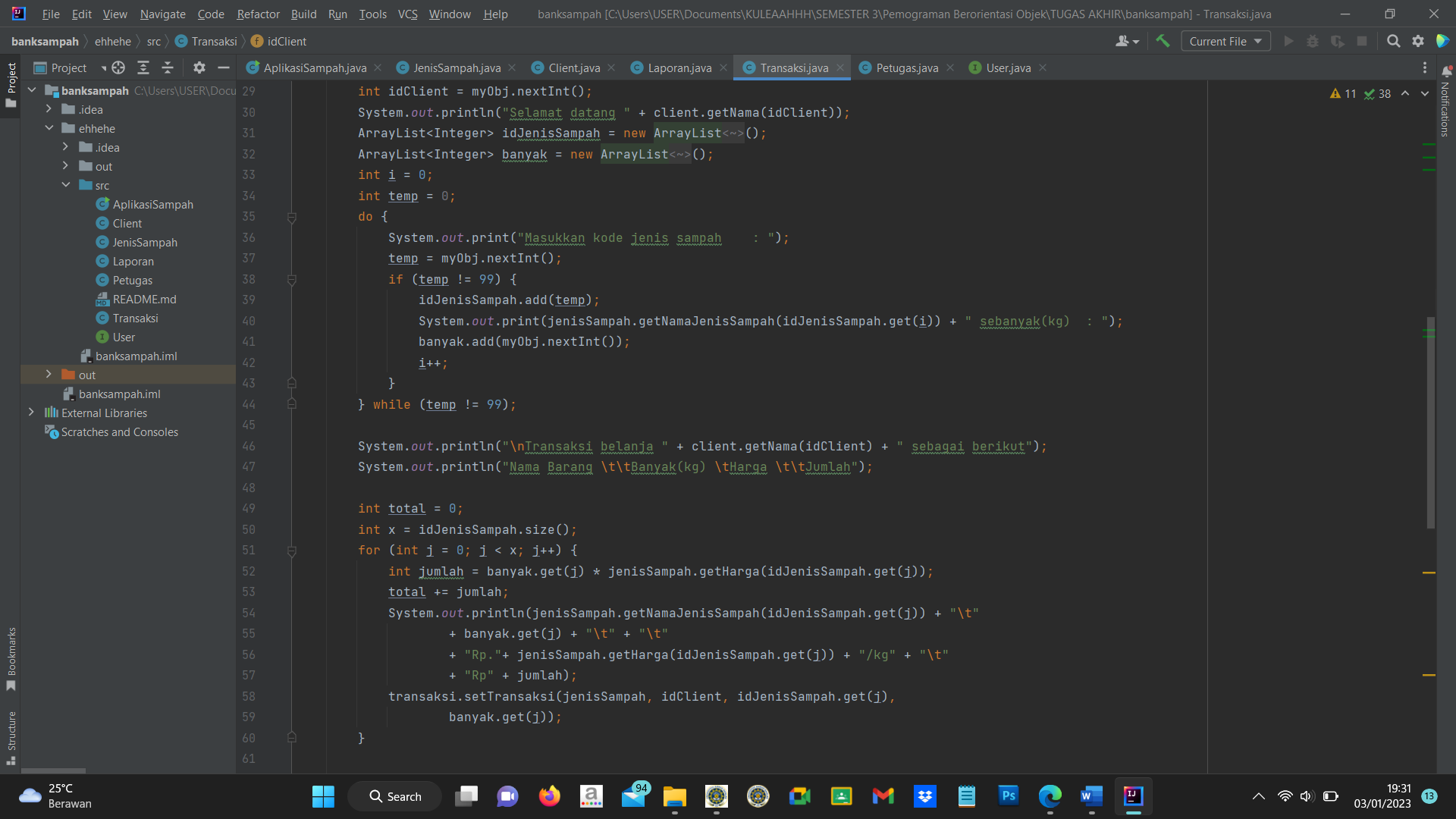
* Laporan.java

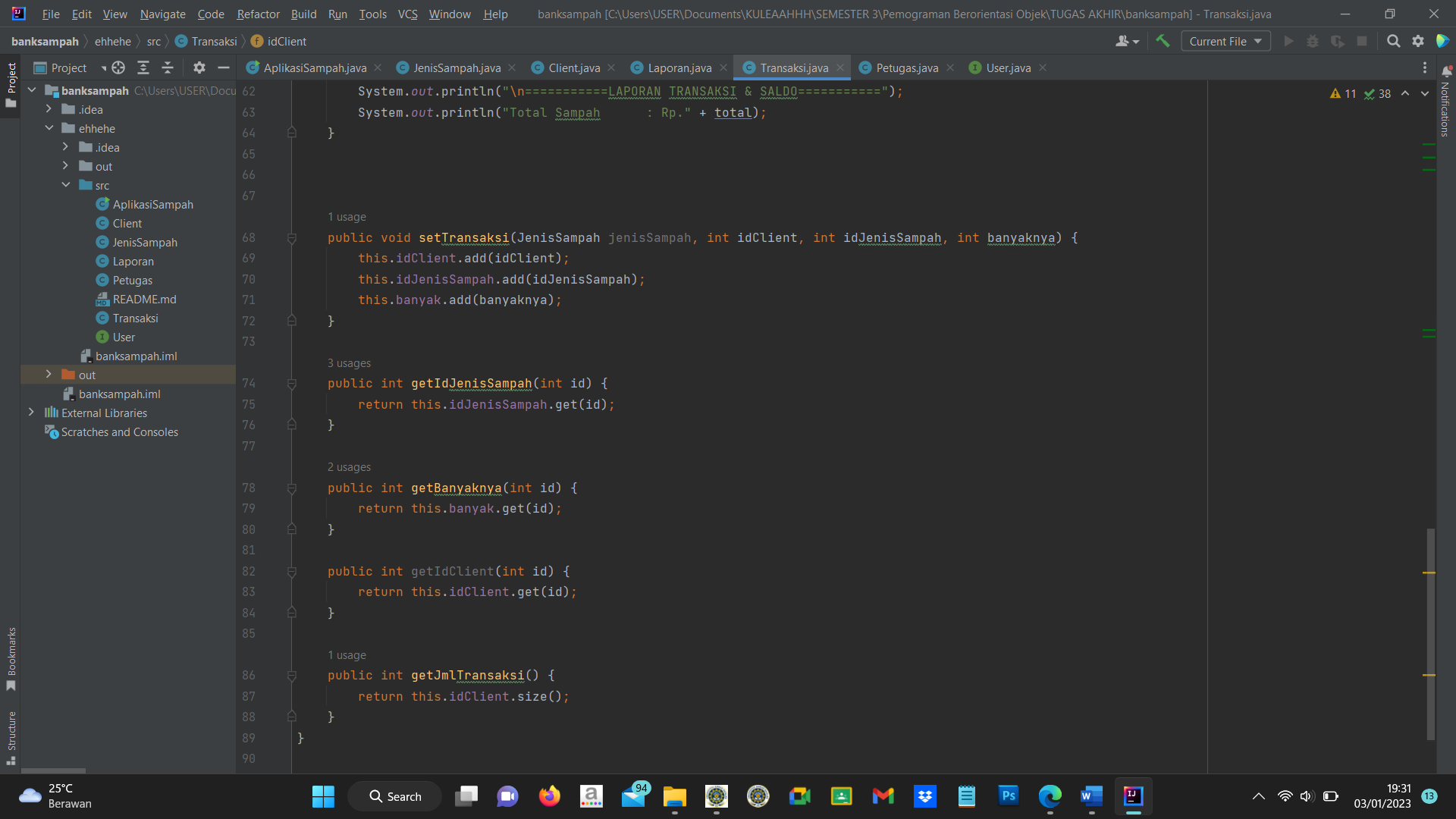




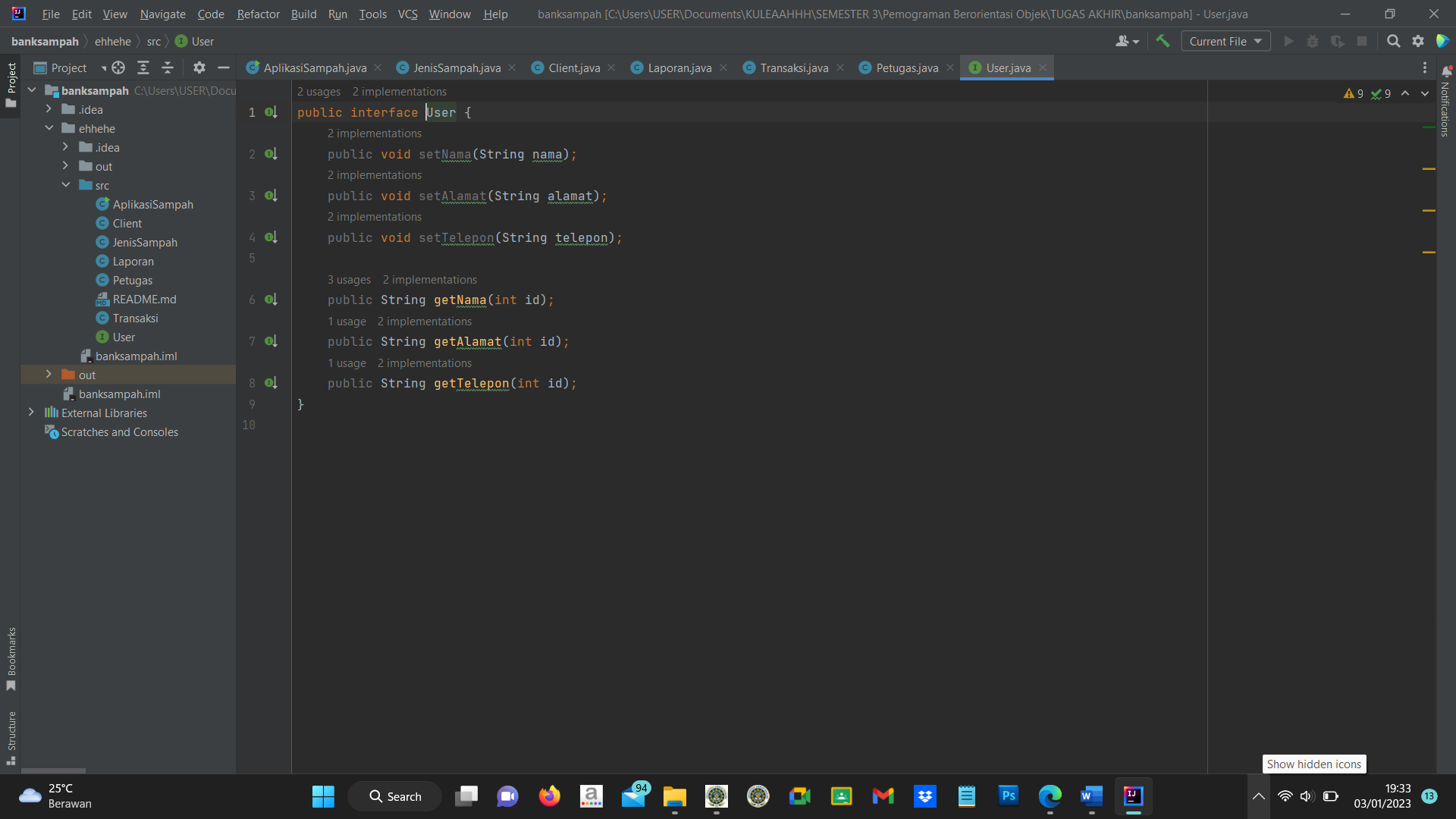
* Transaksi.java







* User.java



* AplikasiSampah.java

public class AplikasiSampah {  
 public static void main(String[] args) {  
 // *TODO code application logic here* JenisSampah jenisSampah1 = new JenisSampah();  
 Client client1 = new Client();  
 Petugas petugas1 = new Petugas();  
 Transaksi transaksi1 = new Transaksi();  
 Laporan laporan1 = new Laporan();  
  
 //Menampilkan jenis sampah dan client  
 laporan1.laporan(jenisSampah1);  
 laporan1.laporan(client1);  
  
 //Proses sampah  
 transaksi1.prosesTransaksi(client1, transaksi1, jenisSampah1);  
  
 //Menampilkan tabel client dan histori  
 laporan1.laporan(transaksi1, jenisSampah1);  
  
  
 }  
}

* JenisSampah.java

import java.util.ArrayList;  
public class JenisSampah {  
  
 private ArrayList<String> namaJenisSampah = new ArrayList<String>();  
 private ArrayList<Integer> harga = new ArrayList<Integer>();  
  
  
 public JenisSampah() {  
 this.namaJenisSampah.add("Jenis Sampah Origanik");  
 this.harga.add(6000);  
 this.namaJenisSampah.add("Jenis sampah Anorganik ");  
 this.harga.add(7500);  
 this.namaJenisSampah.add("Jenis sampah Daur Ulang");  
 this.harga.add(9000);  
 }  
 public int getJmlJenisSampah(){  
 return this.namaJenisSampah.size();  
 }  
 public void setNamaJenisSampah(String namaJenisSampah ){  
 this.namaJenisSampah.add(namaJenisSampah);  
 }  
 public String getNamaJenisSampah(int idJenisSampah){  
 return this.namaJenisSampah.get(idJenisSampah);  
 }  
 public void setHarga(int harga ){  
 this.harga.add(harga);  
 }  
 public int getHarga(int idJenisSampah){  
 return this.harga.get(idJenisSampah);  
 }  
}

* Client.java.

import java.util.ArrayList;  
public class Client implements User{  
  
 private ArrayList<String> namaClient = new ArrayList<String>();  
 private ArrayList<String> alamat = new ArrayList<String>();  
 private ArrayList<String> telepon = new ArrayList<String>();  
 private ArrayList<Integer> saldo = new ArrayList<Integer>();  
  
 public Client() {  
 this.namaClient.add("Arrijal");  
 this.alamat.add("Ketapang");  
 this.telepon.add("08123456666");  
 this.saldo.add(100000);  
  
 this.namaClient.add("Lodri");  
 this.alamat.add("Sandai");  
 this.telepon.add("081222233344");  
 this.saldo.add(200000);  
 }  
 public void setSaldo(int saldo){  
 this.saldo.add(saldo);  
 }  
 public int getSaldo(int idClient){  
 return this.saldo.get(idClient);  
 }  
 public void editSaldo(int idClient, int saldo){  
 this.saldo.set(idClient, saldo);;  
 }  
 public int getJmlClient(){  
 return this.saldo.size();  
 }  
 @Override  
 public void setNama(String namaClient ){  
 this.namaClient.add(namaClient);  
 }  
  
 @Override  
 public void setAlamat(String alamat ){  
 this.alamat.add(alamat);  
 }  
 @Override  
 public void setTelepon(String telepon ){  
 this.telepon.add(telepon);  
 }  
  
 @Override  
 public String getNama(int idClient){  
 return this.namaClient.get(idClient);  
 }  
  
 @Override  
 public String getAlamat(int idClient){  
 return this.alamat.get(idClient);  
 }  
  
 @Override  
 public String getTelepon(int idClient){  
 return this.telepon.get(idClient);  
 }  
}

* Laporan.java

public class Laporan {  
  
 public void laporan(JenisSampah jenisSampah){  
 int x=jenisSampah.getJmlJenisSampah();  
  
 System.*out*.println("\n============TABEL JENIS SAMPAH============");  
 System.*out*.println("Nama Jenis Sampah \tHarga/kg");  
 for (int i = 0; i < x; i++) {  
 System.*out*.println(jenisSampah.getNamaJenisSampah(i)+"\t"+ jenisSampah.getHarga(i) + "/kg");  
 }  
 }  
 public void laporan(Client client){  
 int x=client.getJmlClient();  
  
 System.*out*.println("\n============TABEL CLIENT============");  
 System.*out*.println("Nama \tAlamat \t\tTelepon \tSaldo");  
 for (int i = 0; i < x; i++) {  
 System.*out*.println(client.getNama(i)+"\t"+  
 client.getAlamat(i)+"\t"+client.getTelepon(i)+"\t"+ "Rp." +  
 client.getSaldo(i));  
 }  
 }  
 public void laporan(Transaksi transaksi, JenisSampah jenisSampah){  
 int x=transaksi.getJmlTransaksi();  
  
 System.*out*.println("\n============HISTORI BANK SAMPAH============");  
 System.*out*.println("Nama Jenis Sampah\tJumlah(kg) \tHarga/kg \t\tJumlah");  
  
 int total=0;  
 for (int i = 0; i < x; i++) {  
 int  
 jumlah=transaksi.getBanyaknya(i)\*jenisSampah.getHarga(transaksi.getIdJenisSampah(i));  
 total+=jumlah;  
  
 System.*out*.println(jenisSampah.getNamaJenisSampah(transaksi.getIdJenisSampah(i))+"\t"+  
  
 transaksi.getBanyaknya(i)+ "kg"+ "\t" +"\t"+ "Rp." + jenisSampah.getHarga(transaksi.getIdJenisSampah(i))+ "/kg" + "\t" + "\t"  
 + "Rp."+ jumlah);  
 }  
 System.*out*.println("Total Omset : Rp."+ total);  
 }  
}

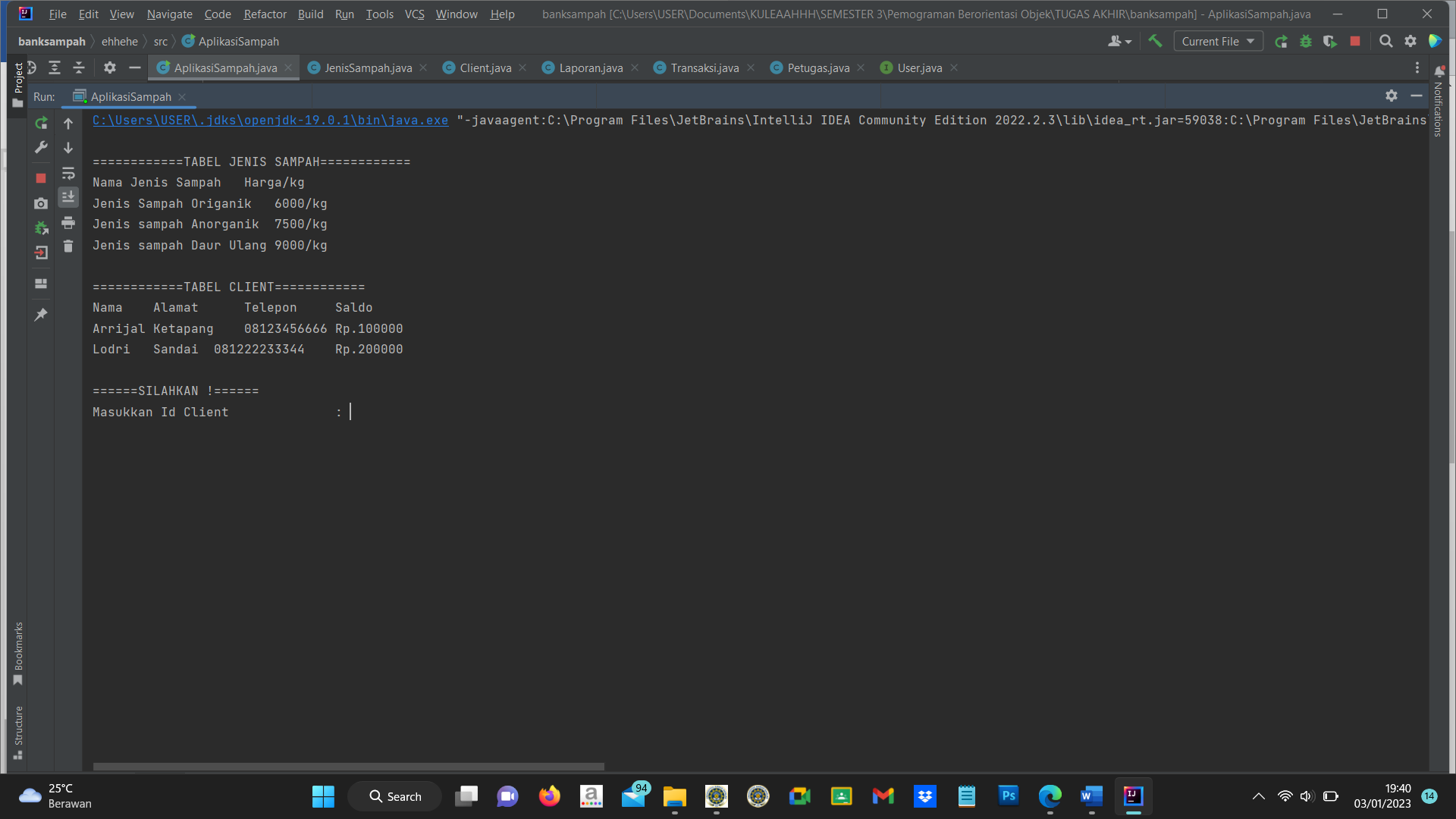
* Transaksi.java

import java.util.ArrayList;  
import java.util.Scanner;  
public class Transaksi {  
  
 private ArrayList<Integer> idJenisSampah = new ArrayList<Integer>();  
 private ArrayList<Integer> banyak = new ArrayList<Integer>();  
 private ArrayList<Integer> idClient = new ArrayList<Integer>();  
  
 public Transaksi() {  
 this.idClient.add(0);  
 this.idJenisSampah.add(0);  
 this.banyak.add(2);  
 this.idClient.add(0);  
 this.idJenisSampah.add(1);  
 this.banyak.add(3);  
 this.idClient.add(1);  
 this.idJenisSampah.add(0);  
 this.banyak.add(1);  
 this.idClient.add(1);  
 this.idJenisSampah.add(2);  
 this.banyak.add(2);  
 }  
  
 public void prosesTransaksi(Client client, Transaksi transaksi, JenisSampah jenisSampah) {  
 Scanner myObj = new Scanner(System.*in*);  
 System.*out*.println("\n======SILAHKAN !======");  
 System.*out*.print("Masukkan Id Client : ");  
 int idClient = myObj.nextInt();  
 System.*out*.println("Selamat datang " + client.getNama(idClient));  
 ArrayList<Integer> idJenisSampah = new ArrayList<Integer>();  
 ArrayList<Integer> banyak = new ArrayList<Integer>();  
 int i = 0;  
 int temp = 0;  
 do {  
 System.*out*.print("Masukkan kode jenis sampah : ");  
 temp = myObj.nextInt();  
 if (temp != 99) {  
 idJenisSampah.add(temp);  
 System.*out*.print(jenisSampah.getNamaJenisSampah(idJenisSampah.get(i)) + " sebanyak(kg) : ");  
 banyak.add(myObj.nextInt());  
 i++;  
 }  
 } while (temp != 99);  
  
 System.*out*.println("\nTransaksi belanja " + client.getNama(idClient) + " sebagai berikut");  
 System.*out*.println("Nama Barang \t\tBanyak(kg) \tHarga \t\tJumlah");  
  
 int total = 0;  
 int x = idJenisSampah.size();  
 for (int j = 0; j < x; j++) {  
 int jumlah = banyak.get(j) \* jenisSampah.getHarga(idJenisSampah.get(j));  
 total += jumlah;  
 System.*out*.println(jenisSampah.getNamaJenisSampah(idJenisSampah.get(j)) + "\t"  
 + banyak.get(j) + "\t" + "\t"  
 + "Rp."+ jenisSampah.getHarga(idJenisSampah.get(j)) + "/kg" + "\t"  
 + "Rp" + jumlah);  
 transaksi.setTransaksi(jenisSampah, idClient, idJenisSampah.get(j),  
 banyak.get(j));  
 }  
  
 System.*out*.println("\n===========LAPORAN TRANSAKSI & SALDO===========");  
 System.*out*.println("Total Sampah : Rp." + total);  
 }  
  
  
  
 public void setTransaksi(JenisSampah jenisSampah, int idClient, int idJenisSampah, int banyaknya) {  
 this.idClient.add(idClient);  
 this.idJenisSampah.add(idJenisSampah);  
 this.banyak.add(banyaknya);  
 }  
  
 public int getIdJenisSampah(int id) {  
 return this.idJenisSampah.get(id);  
 }  
  
 public int getBanyaknya(int id) {  
 return this.banyak.get(id);  
 }  
  
 public int getIdClient(int id) {  
 return this.idClient.get(id);  
 }  
  
 public int getJmlTransaksi() {  
 return this.idClient.size();  
 }  
}

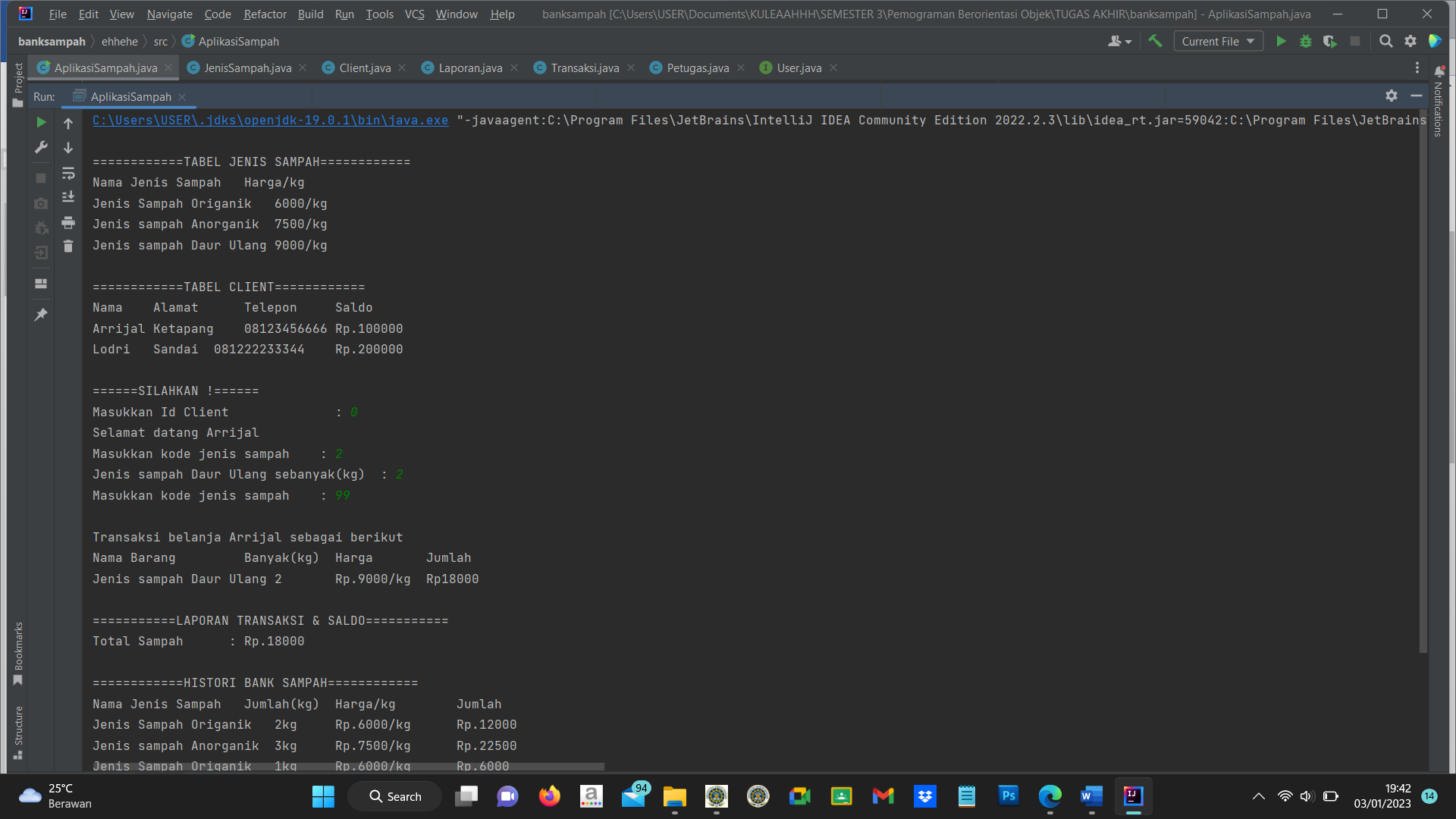
* User.java

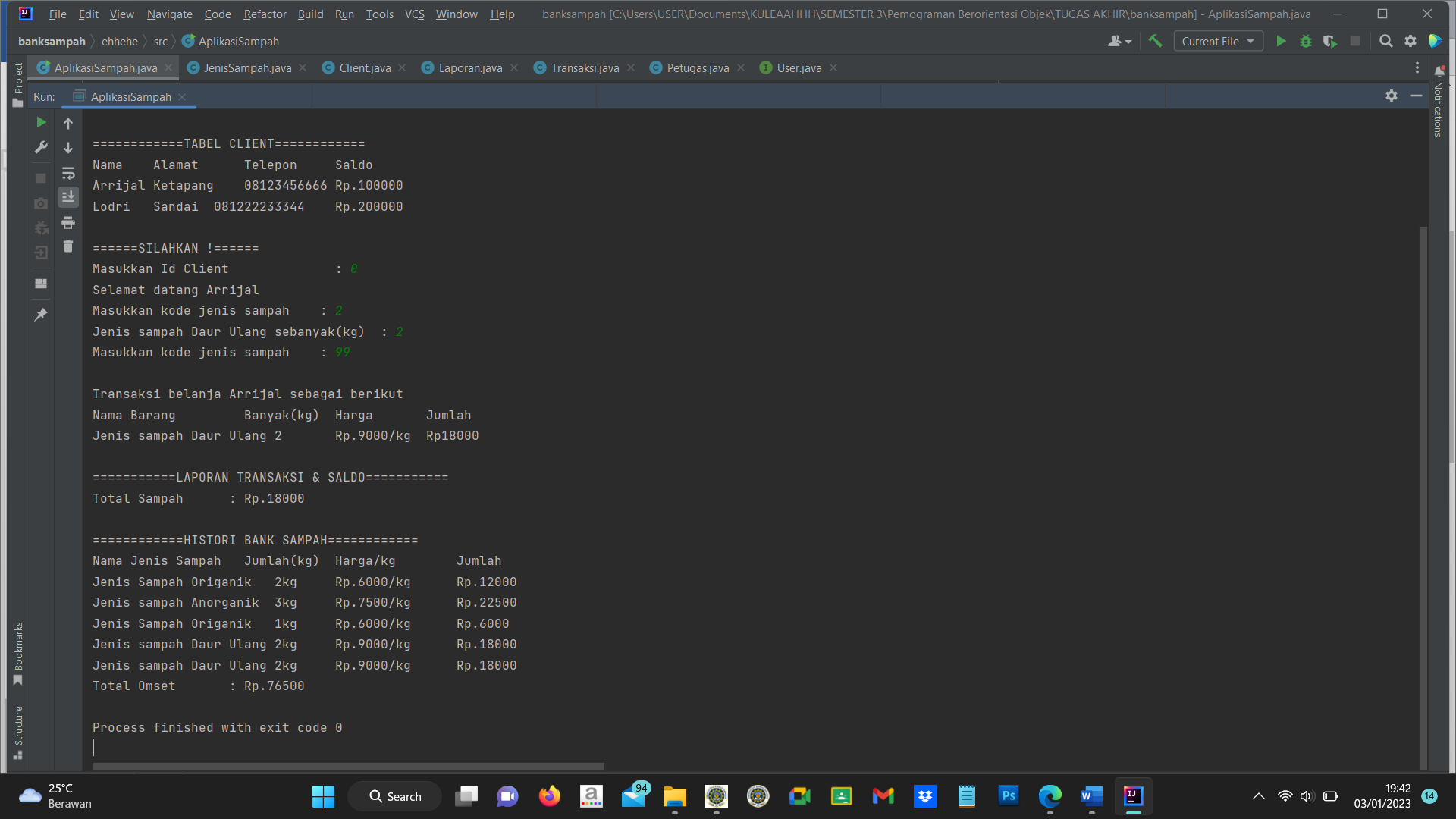
public interface User {  
 public void setNama(String nama);  
 public void setAlamat(String alamat);  
 public void setTelepon(String telepon);  
  
 public String getNama(int id);  
 public String getAlamat(int id);  
 public String getTelepon(int id);  
}

1. Penjelasan screenshot tampilan yang dihasilkan aplikasi.



Uji Coba program beserta capture eror/sukses.





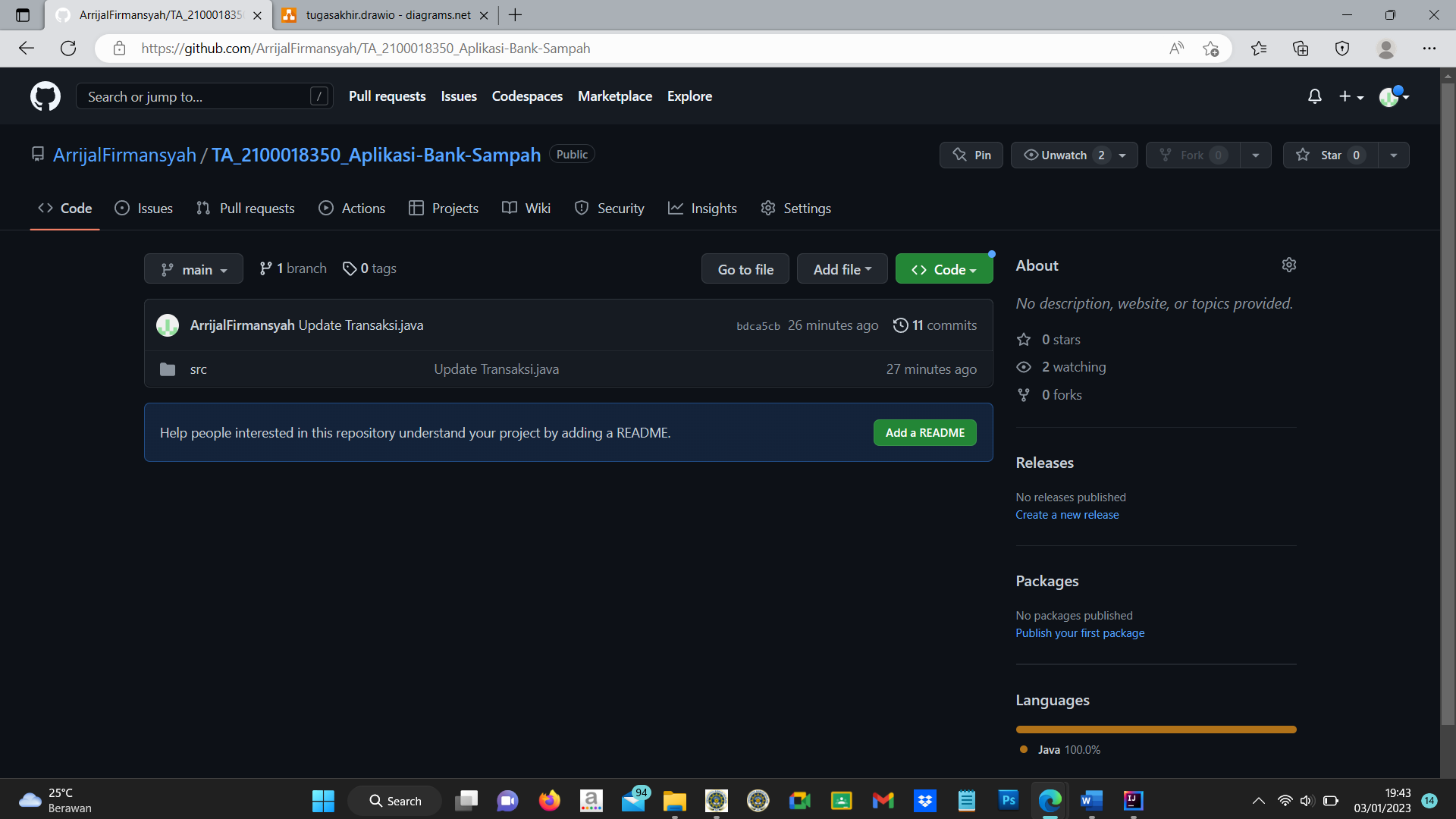
Penjelasan :

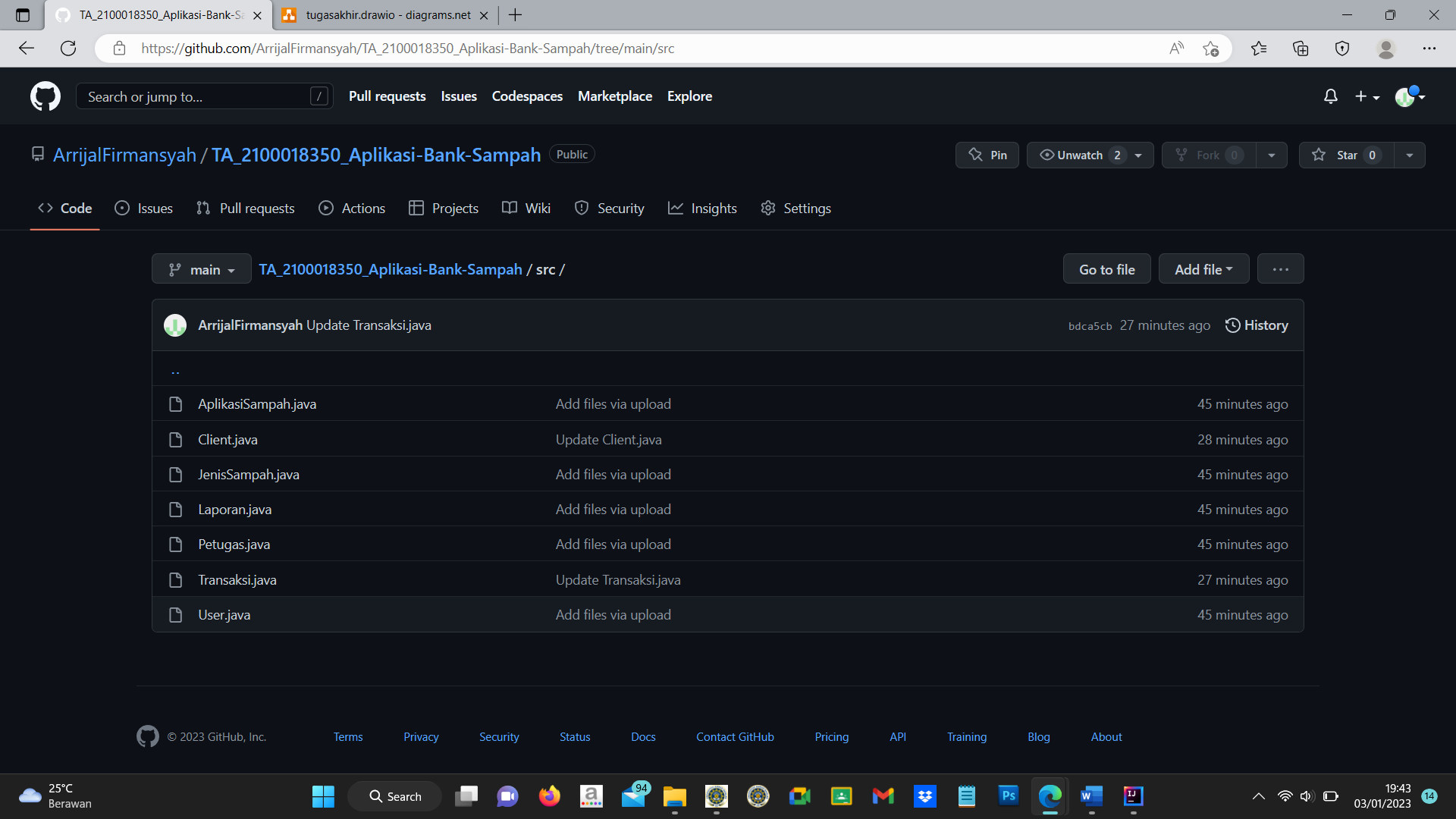
Program aplikasi ini dirancang untuk nantinya akan digunakan untuk menukarkan barang-barang yang tidak terpakai (sampah) Organik, daur ulang dan sampah Anorganik. Berdasarkan kriteria sampah yang ditukarkan maka user akan mendapatkan uang/hadiah yang berbeda-beda juga tergantung dari berat (kg) dan kriteria jenis sampah yang ditukarkan.

Pada pertamakali menjalankan user akan mendapatkan TABEL JENIS SAMPAH dan TABEL CLIENT setelah mendapatkan kedua tabel user akan diminta untuk menginputkan Id client. Setelah menginput Id client user akan diminta untuk memasukkan kode jenis sampah. Disini ada 3 pilihan untuk 0 = jenis sampah organic, 1=Jenis sampah anorganik dan 2=Jenis sampah daur ulang dan proses penginputan selesai dengan memasukkan nomor 99 Setelah itu lanjut ke proses transaksi penukaran yang nani akan berisi jenis sampah, banyak(kg), harga(kg) dan jumlah semuahnya.kemudian selesai.

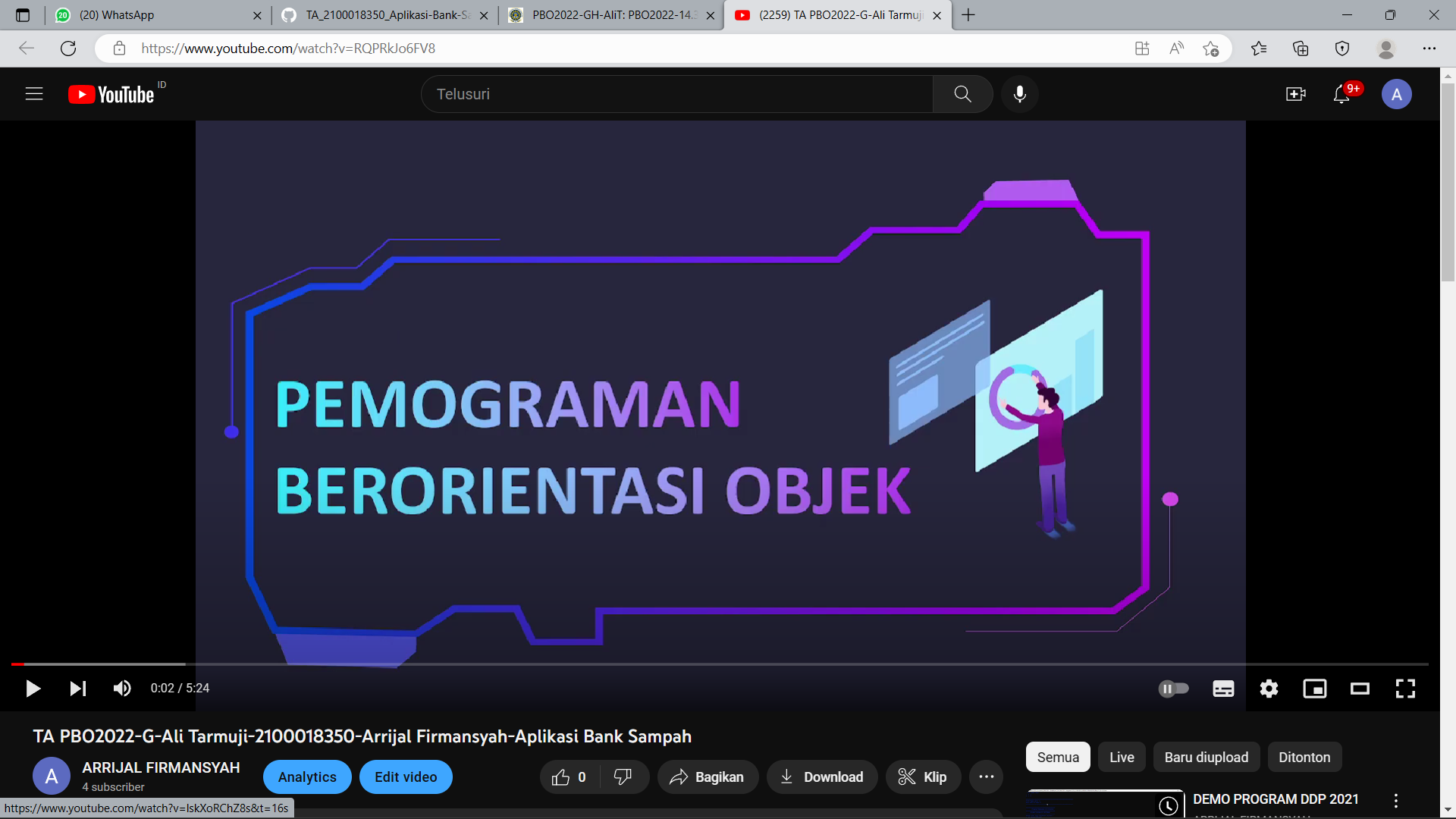
1. Penjeleasan screenshot status unggah skrip di Gitlab/Github hingga projek final.

LINK : <https://github.com/ArrijalFirmansyah/TA_2100018350_Aplikasi-Bank-Sampah/tree/main/src>



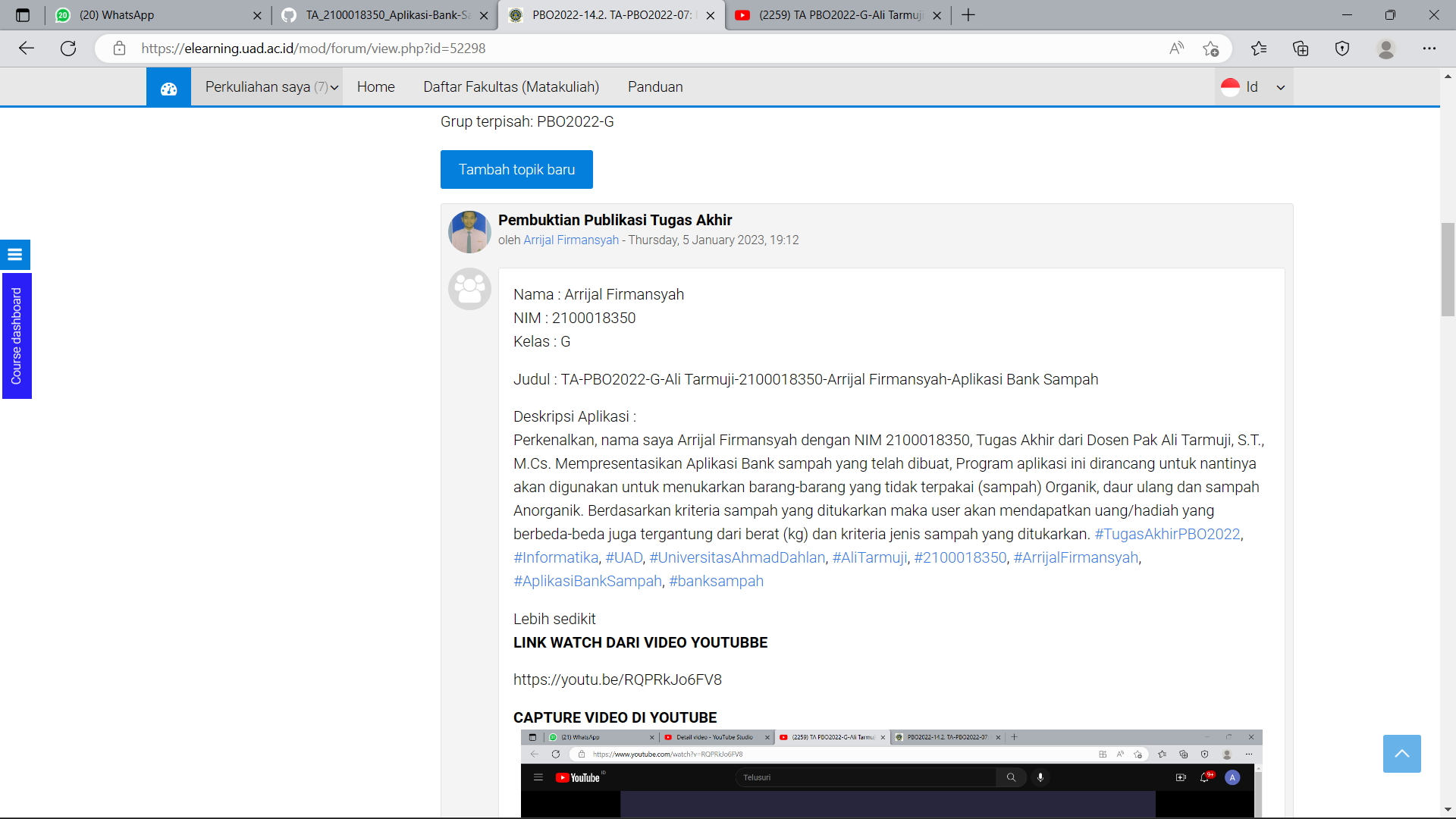


1. Penjelasan screenshot status publish di Youtube dan forum Elearning.
2. Screenshot tampilan publikasi di youtube (Fullscreen browser)



LINK YOUTUBE : <https://youtu.be/RQPRkJo6FV8>

1. Screenshot bukti publikasi di forum (tugas sebelumnya) capture fullscreen browser



1. Analisa pengerjaan projek (tinjauan dari sisi waktu, ketercapaian spesifikasi, biaya yang dibutuhkan, kendala, tantangan masa depan, dan atau lain-lain jika ada).
2. Waktu.

Estimasi waktu yang saya butuhkan untuk membuat aplikasi ini sekitar 1 – 2 minggu atau lebih tepatnya 12 hari.

1. Ketercapaian spesifikasi.

Ketercapaian spesifikasi dari kode yang saya buat adalah lebih mudah dipahami untuk diri sendiri tidak banyak sintak dari luar atau lebih tepatnya berada pada ruang lingkup materi yang di ajarkan.

1. Biaya yang dibutuhkan.

Anggaran yang mungkin dibutuhkan adalah biaya listrik dan kuota internet, karena laptop yanbg saya gunakan tentunya akan tertap hidup, yang pastinya membutuhkan daya listrik.

1. Kendala.

Ada beberapa kendala yang saya alami pada saat membuat aplikasi ini mulai dari tidak efisien dalam mengerjakan dan mengubah konsep diagaram serta kode yang ada didalam nya, dan pada github saya juga mengalami kendala yaitu lupa password.

1. Tantangan masa depan.

Mungkin untuk kedepannya agar suatu proyek bisa maksimal, tentu waktu yang diberikan harus di efiseinsi kan dengan baik dan penuh dengan rasa pensaran akan aplikasi yang kita buat.